

ПЛАТФОРМА URANIE ДЛЯ КВАНТИФІКАЦІЇ НЕВИЗНАЧЕНОСТІ

Горбачук В.М., Бардадим Т.О., Дунаєвський М.С., Голоцукова Т.Г., Рибачок Д.О.

*Відділ інтелектуальних інформаційних технологій,
Інститут кібернетики імені В.М.Глушкова НАН України
GorbachukVasyl@netscape.net, VGorbachuk@nas.gov.ua*

Платформа ПЗ Uranie – це фреймворк ПЗ з відкритим вихідним кодом, розроблений у підрозділі ядерної енергії СЕА (Commissariat à l'énergie atomique et aux énergies alternatives; Alternative Energies and Atomic Energy Commission; заснований Францією 18 жовтня 1945 р. після перемоги у Другій світовій війні; перший директор – Фредерік Жоліо-Кюрі, Нобелівський лауреат з хімії 1935 р.). Uranie призначена для кількісного оцінювання невизначеності, дослідження чутливості, генерування сурогатної моделі (мета-моделі), калібрування кодів, щоб сприяти аналізу та посиленню достовірності результатів комплексного моделювання [1]. ПЗ Uranie застосовує технології С++ та програмного фреймворку ROOT, розробленого в CERN для обробки великих обсягів даних [2], що пропонує інтерфейси Python і може бути встановлений через Anaconda.

Система ROOT – це об'єктно-орієнтований фреймворк з відкритим кодом для широкомасштабного аналізу даних, який розпочався як приватний проект у 1995 р. в CERN і перетворився на офіційно підтримуваний інструментарій аналізу для великого адронного коллайдера (Large Hadron Collider (LHC)).

Uranie включає досить універсальні (generic) засоби для генерування планування експериментів (design-of-experiments) [3], створення сурогатних моделей для імітації коду, навчання сурогатних моделей (наприклад, через поширення невизначеностей та поліноміальне розкладання хаосу (polynomial chaos expansion), нейронні мережі, процес Гаусса (крігінг (kriging))), різних методів аналізу чутливості, алгоритмів оптимізації, ефективної глобальної оптимізації (efficient global optimization (EGO)), спроможностей паралельних обчислень, трактуючи зовнішні коди моделювання у багатьох галузях фізики (наприклад, теплообмін між пластинчатим листом (plate-sheet) і рідиною) як деякі чорні скриньки (black boxes) [4].

Тому Uranie виграє від численних особливостей ROOT через інтерактивний інтерпретатор С++ (Cling), побудований на основі LLVM і Clang, інтерфейс Python (PyROOT), доступ до баз даних SQL тощо. Uranie виграє від спеціального використання зовнішніх і внутрішніх факторів, зокрема від ефективної моделі обробки даних, передових графічних засобів (наприклад, на веб-сторінці Sourceforge <https://sourceforge.net/projects/uranie/>, де можна отримати також вихідний код платформи Uranie), певних рішень для паралелізації.

Після отримання вихідних кодів рекомендується виконати інсталяцію згідно з файлом README, яка також дасть документацію Uranie (методичний посібник (у форматах html, pdf) з попереднім вступом до основних (математичних) методів і алгоритмів і додатковими посиланнями для поглибленого розуміння проблематики), посібник користувача (у форматах html, pdf) з поясненнями щодо реалізацій методів разом і численними прикладами, посібник розробника з описом (обчислювальних) методів на платформі Doxygen).

Doxygen – це генератор (generator) документації (documentation), який працює з багатьма мовами програмування. Він витягує інформацію зі спеціально відформатованих коментарів до вихідного коду та зберігає її в одному з різних підтримуваних форматів. Doxygen підтримує статичний аналіз кодової бази, використовуючи дерево (синтаксичного) розбору, яке автоматизовано виводить з кодової бази для генерування діаграм і блок-схем структури коду. Doxygen забезпечує перехресні посилання, які читач може використовувати для повернення до вихідного (оригінального) коду зі згенерованої документації.

Doxygen можна використовувати в багатьох контекстах програмування. Він підтримує багато мов, включаючи С, С++, С#, D, Fortran, IDL, Java, Objective-C, Perl, PHP, Python, VHDL, і може працювати на багатьох комп'ютерах, включаючи Unix-подібні системи, macOS та

Windows. Doxygen – це вільне ПЗ, випущене відповідно до умов GNU General Public License версії 2 (GPLv2).

Перша версія Doxygen запозичила код з ранньої версії DOC++, яку розробили Роланд Вундерлінг (здобув науковий ступінь доктора філософії з математики Технічного університету Берліна (започаткованого у 1770 р.) у 1997 р.; відомий як архітектор CPLEX (пакет ПЗ (розв'язувач), призначений для розв'язування задач лінійного та квадратичного програмування, зокрема цілочисельного; назва CPLEX походить від симплекс-методу, реалізованого на мові програмування С, хоча зараз пакет надає різні методи оптимізації з інтерфейсом на інших мовах програмування; комерційна версія вийшла у 1988 р.)) та Мальте Цюклер (здобув науковий ступінь магістра в Інституті астрономії та астрофізики Технічного університету Берліна у 1997 р.) в Інституті Цузе Берліна (Zuse Institute Berlin (ZIB)). Пізніше код Doxygen переписав Дімітрі ван Хееш (здобув науковий ступінь бакалавра з менеджменту інформації Університету Тілбурга (заснованого у 1927 р.) у 2006 р.).

ZIB (Konrad-Zuse-Zentrum für Informationstechnik Berlin) – це дослідницький інститут прикладної математики та інформатики в кампусі Вільного університету Берліна в Далемі (Берлін, Німеччина) (заснований у Західному Берліні у 1948 р.). ZIB був заснований законом як статутна установа та як неуніверситетський дослідницький інститут землі Берлін у 1984 р. ZIB, у тісній міждисциплінарній співпраці з університетами та науковими установами Берліна, здійснює дослідження і розробки в галузі інформаційних технологій з особливим акцентом на прикладну алгоритмічну математику та практичні комп'ютерні науки. ZIB також надає високопродуктивні комп'ютерні потужності як супутню послугу в рамках Мережі високопродуктивних комп'ютерів у Північній Німеччині (Norddeutscher Verbund von Hoch- und Höchstleistungsrechnern (HLRN)).

Конрад Цузе, на честь якого названо ZIB, народився у м. Берлін Німецької імперії у 1910 р., здобув кваліфікацію інженера Берлінської вищої технічної школи в Шарлоттенбурзі (Технічного університету Берліна) у 1935 р., помер у м. Хюфельд землі Гессен Німеччини у 1995 р. Цузе збудував перший у світі програмований комп'ютер; функціональний програмно-керований комп'ютер Z3, повний за Тьюрінгом, почав працювати в травні 1941 р. Завдяки цій машині та її попередникам Z1 та Z2, Цузе вважається винахідником і батьком сучасного комп'ютера. Z2 був електромеханічним (механічним та релейним) цифровим комп'ютером, який був збудований у 1940 р. і вдосконаливав Z1 з механічною пам'яттю.

Високопродуктивні (high-performance) обчислювальні ресурси, постійне поліпшення точності чисельного моделювання та відповідних експериментальних вимірювань потребують посилення довіри до результатів цього моделювання, зокрема шляхом кількісного оцінювання (квантифікації) невизначеності. Таке оцінювання може залежати від запитуваних цілей (ітеративне ранжування джерел невизначеностей через аналіз чутливостей, зниження невизначеностей в обчислювальних прикладах і практичних застосуваннях, точне оцінювання критичного порогу чи оптимального робочого рівня), потребуючи математичних методів тієї чи іншої складності [5].

Традиційно квантифікація вимагала великої кількості запусків коду для отримання надійних результатів, тобто суттєвих обчислювальних і часових ресурсів. Щоб долати цей недолік, проводилися відповідні методологічні та технічні розробки. При зростанні ресурсів, доступних для складних обчислень, наприклад, еволюції рідини на дрібній сітці (fine mesh), фізикам корисно знати, наскільки дрібною має бути ця сітка. Замість того, щоб збільшувати роздільну здатність сітки, може бути кращим зменшувати конкретне джерело невизначеності чи додавати нові розташування до навчальної бази даних для побудови сурогатної моделі.

Платформа Uranie була розроблена для того, щоб збирати методологічні розробки, які надходять з промислового й академічного світу, та надавати їх якнайширшій аудиторії завдяки кільком важливим аспектам: платформа з відкритим вихідним кодом є доступною для всіх, а кожний мотивований користувач може досліджувати такий код і пропонувати його вдосконалення чи виправлення; платформа розроблена на Linux з метою її доступності, надаючи можливість Windows-портування. Щоб перевіряти та гарантувати найкращу портабельність, платформа щодня тестується на семи різних дистрибутивах Linux і Windows;

платформа написана на C++ з можливістю її використання через Python; платформа організована за модульним принципом, щоб завантажувати лише потрібні модулі, організуючи аналіз через компіляцію фундаментальних блоків (bricks); платформа є універсальною, працюючи як з явною функцією, так і з деяким кодом, вважаючи його чорною скринькою (якщо є можливим обмін даними з цим кодом, скажімо, через файл). Така універсальність збільшує сферу застосовності Uranie.

Ядро платформи Uranie – це модуль бібліотеки DataServer (сервер даних), який містить усю необхідну інформацію про змінні задачі (наприклад, назви, одиниці вимірювання, закони ймовірності, файли даних тощо) та власне дані (якщо інформація була надіслана чи згенерована), дозволяючи виконувати найпростіші статистичні операції (обчислення середніх значень, стандартних відхилень, квантилів і так далі).

Модуль бібліотеки Sampler дозволяє створювати велике розмаїття планів експериментів (вибірок) [3] залежно від даної задачі (поширення невизначеностей, побудова сурогатної моделі тощо). Деякі методи створення планів експериментів загалом вбудовуються у складніші методи, наприклад, у методи у спряженому просторі Фур'є. Модуль бібліотеки Modeler дозволяє будувати одну або кілька сурогатних моделей, які забезпечують простішу і швидшу функцію для емуляції заданого виходу Y складної моделі (яка зазвичай потребує багато ресурсів) для заданого набору вхідних факторів $X_i, i = 1, \dots, n_X$.

Модулі бібліотек Optimizer та Reoptimizer призначені для оптимізації та калібрування моделі, яке полягає у встановленні ступеня свободи моделі таким чином, щоб імітації були якнайближчими до експериментальної бази даних. Оптимізація вважається складною процедурою, оскільки виконання однокритеріального чи багатокритеріального аналізу з обмеженнями чи без умов спирається на певні локальні або глобальні підходи.

Модуль бібліотеки Sensitivity (чутливості) дозволяє виконувати аналіз чутливості одного або кількох вихідних відгуків Y коду відносно вибраних вхідних факторів $X_i, i = 1, \dots, n_X$. Застосовуються скринінговий метод аналізу чутливості (Morrisa) та різні оцінки коефіцієнтів Соболя.

Список літератури

1. The Uranie Team. *User manual for Uranie v 4.10.0. C++ version*. СЕА, 2025. 773 p.
2. Горбачук В.М., Беспалов С.А., Гладковський В.В., Пустовойт М.М. Від електронних документів ЦЕРН до великих даних. *Глушковські читання* (26 жовтня 2023 р., Київ, Україна). Київ: Інститут кібернетики імені В.М.Глушкова НАН України, 2023. С. 27–30.
3. Dragalin V. Model-assisted adaptive designs for dose-ranging clinical trials. *International Conference dedicated to the 60th anniversary of the foundation of Vladimir Andrunachievici Institute of Mathematics and Computer Science* (October 10–13, 2024, Chisinau, Moldova). I. Titchiev, S. Cojocar, F. Damian, V. Izbas, C. Gaidric, D. Lozovanu, N. Vulpe (eds.). Chisinau, Moldova: Vladimir Andrunachievici Institute of Mathematics and Computer Science; Moldova State University, 2024. P. 8–13.
4. Bardadym T.O., Gorbachuk V.M., Novoselova N.A., Osypenko S.P., Skobtsov V.Yu. Intelligent analytical system as a tool to ensure the reproducibility of biomedical calculations. *Штучний інтелект*. 2020. № 3. P. 67–81.
5. Gorbachuk V., Dunaievskiy M., Suleimanov S.-B. Cybersecurity investments in networks. *Stochastic Modeling and Optimization Methods for Critical Infrastructure Protection, Volume 2: Methods and Tools*. A.A. Gaivoronski, P.S. Knopov, V.I. Norkin, V.A. Zaslavskiy (eds.). Wiley-ISTE, 2025. P. 211–234.